FINAL REPORT
OF TEAM PROJECT B

Team Name : HELLO GIRLS

Date :Jan 1,2013

Contents

Ⅰ Prototype System Introduction

§ 1.1 Functions

segmentation.py is a program to segregate Chinese sentences.

§ 1.2 Running Environment

Windows 7

§ 1.3 Developing Environment

Python3.2.3 PyScripter2.5.3

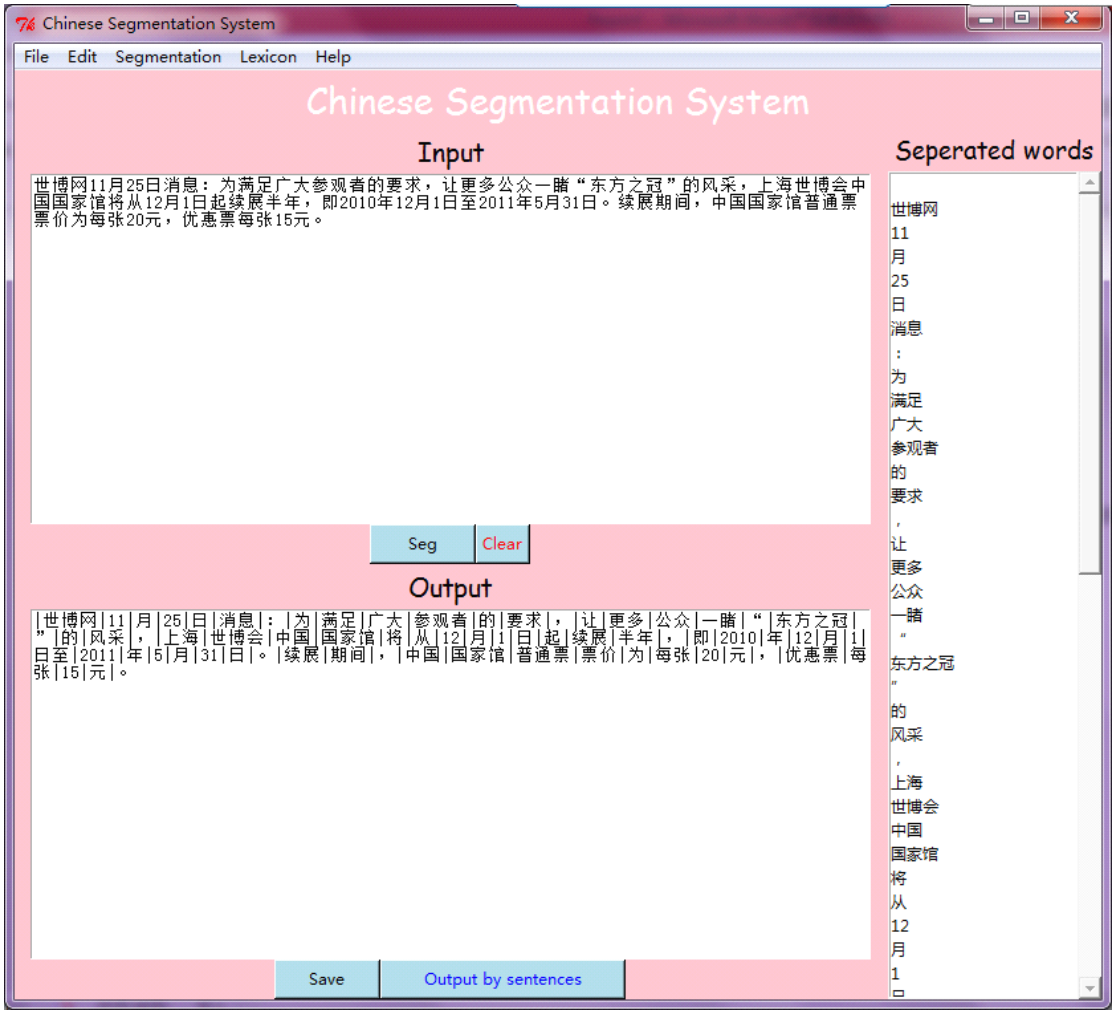
Ⅱ Task Allocation

Name	Team Name: Hello Girls	Task:
Member1: 毛雪娇	**Student No.** 5120309668 **Email:** 377689782@qq.com	**User Interface**
Member2: 王 玫	**Student No.** 5120309691 **Email:** 594189534@qq.com	**Implementation ,Debugging System Testing , Making Report and PPT**
Member3: 何 琨	**Student No.**5120309693 **Email:** 297422878@qq.com	**Lexicon Construction**
Leader : 刘静静	**Student No.** 5120309669 **Email:** 385858021@qq.com	**System Architecture, Component, and Word Segmentation Algorithm**

III User Interface Component

§ 3.1 The Main Interface

The interface appearance:



① Title

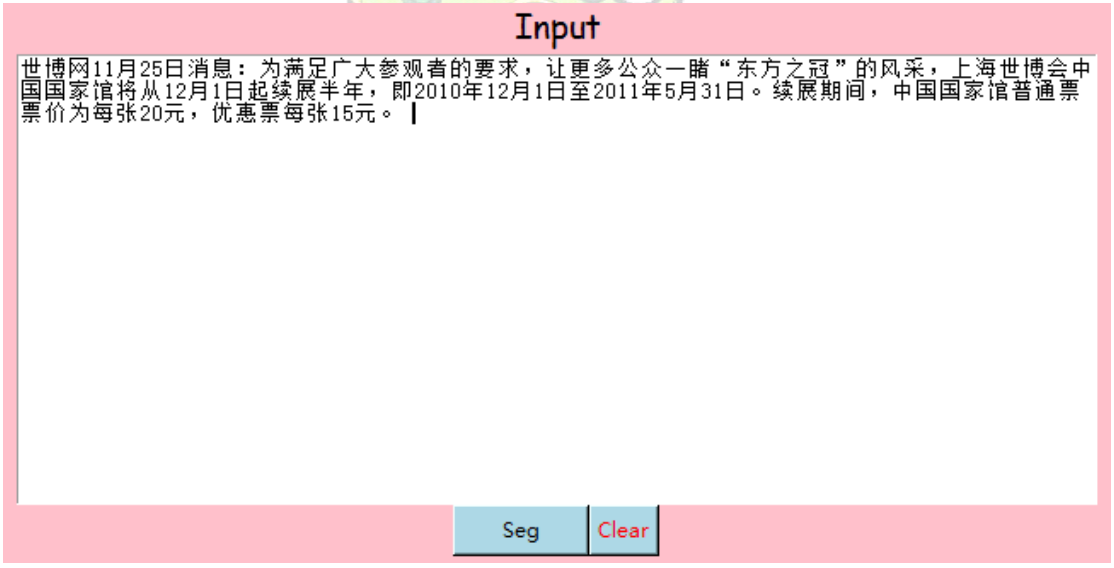
② Input

a) When you open a file, the contents will be shown in this part.

b) You can entry sentences or clear it word by word by yourself in this section.

c) You can stick text coped from other articles into this section.

d) There are two buttons below the entry. One is “seg”, by which you run the main segmentation procedure. And the other is “clear”. If you click it, you will clear the contents of *Input*, *Output* and *Wordlist* at one go.



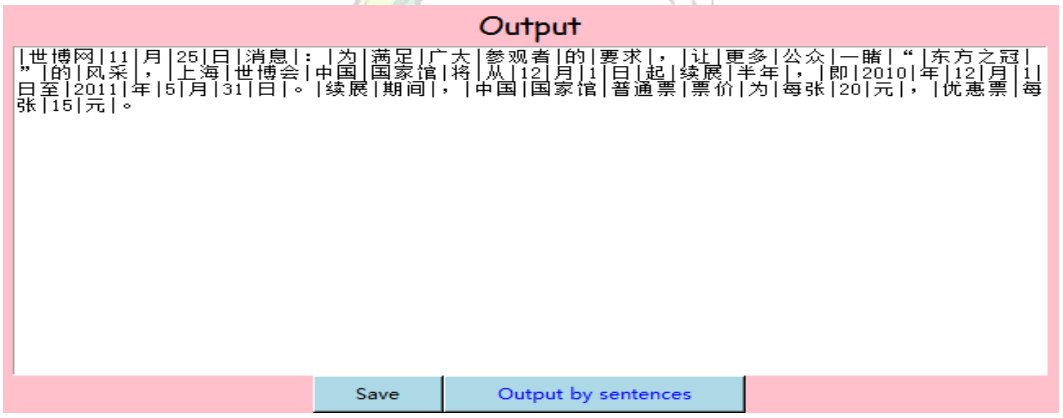
③ Output

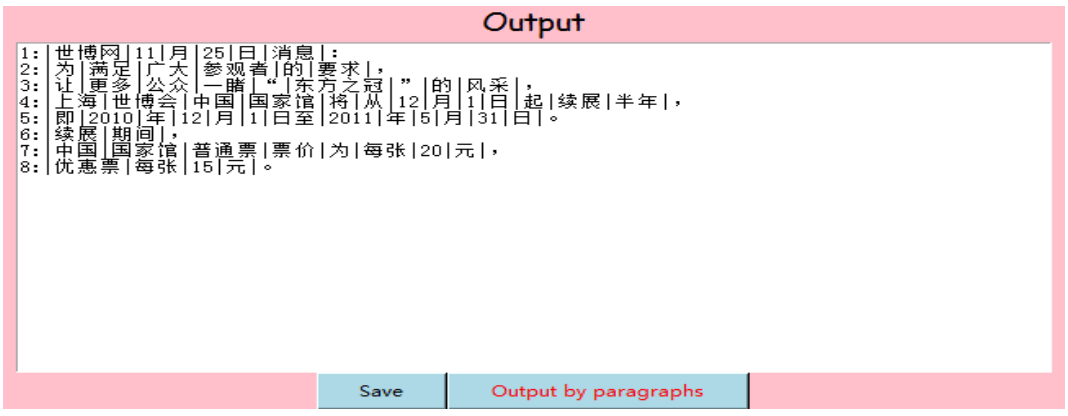
a) The result of segmentation will be presented in this section.

b) You add or delete words in it if you want to.

c) You can save the result into a new file(*.txt) when click the button *save*, and it works the same way as the save option in the menu "file", which will be introduced later.

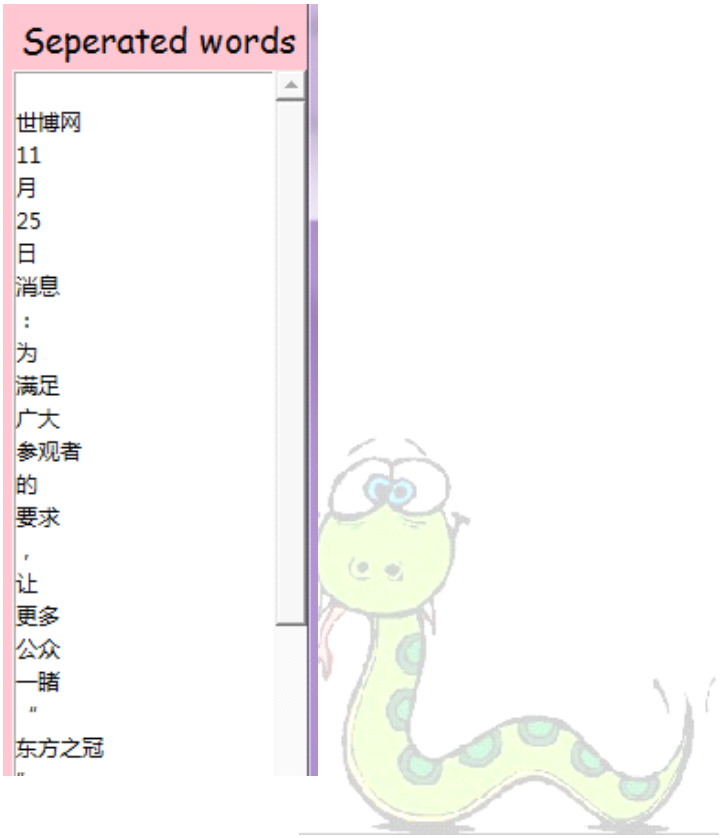
d) We have a convenient button here. It's a changeable button. At first , it shows the text--*Output by sentences* and the contents is the first picture followed. When you click it, the text of button change to—*Output by paragraphs*, and the contents is shown in the second picture. Every you click, it will change to the text of button when before you click





④ Wordlist:

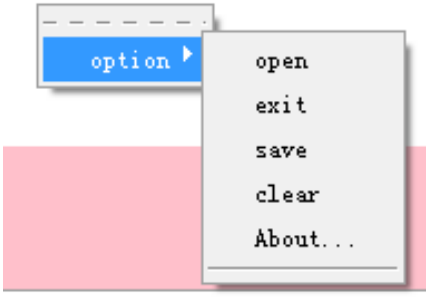
It will show the seperated words one in a row, with a scrollbar if the list is longer than the main window.



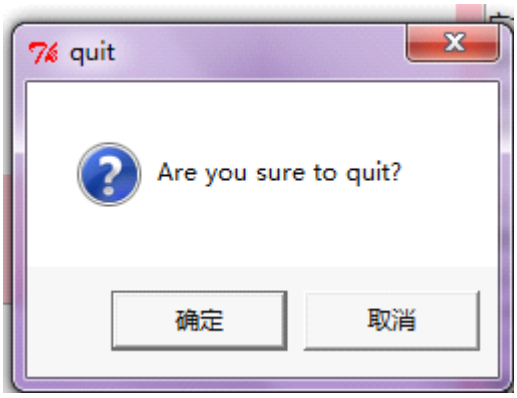
⑤ Right click

We include a right click in our interface for convenience.

“open”, “exit” ,“save” ,“clear”, “About…” have the same functions as those in menu and the main interface.

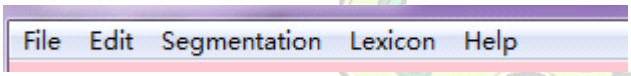


⑥ If you want to quit, in any way, you will receive this warning box! Just click "确定" to quit the program and click "取消" to return!

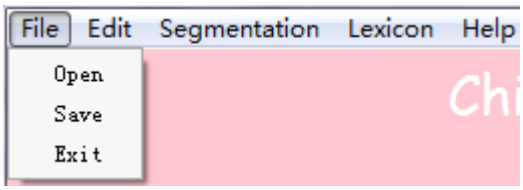


§ 3.2 Five Menus

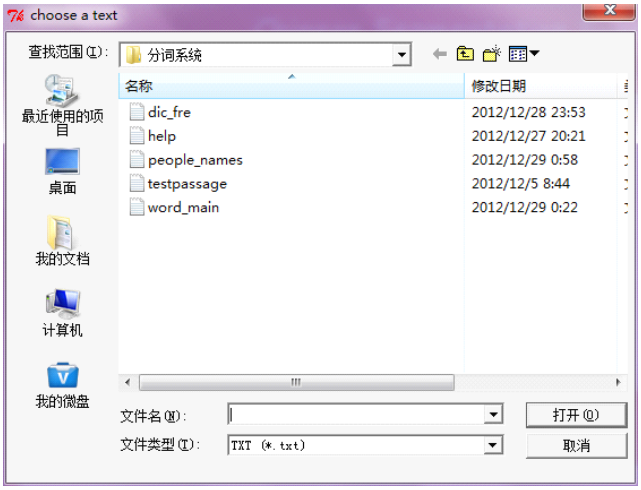
The menu bar consists of five different submenus.



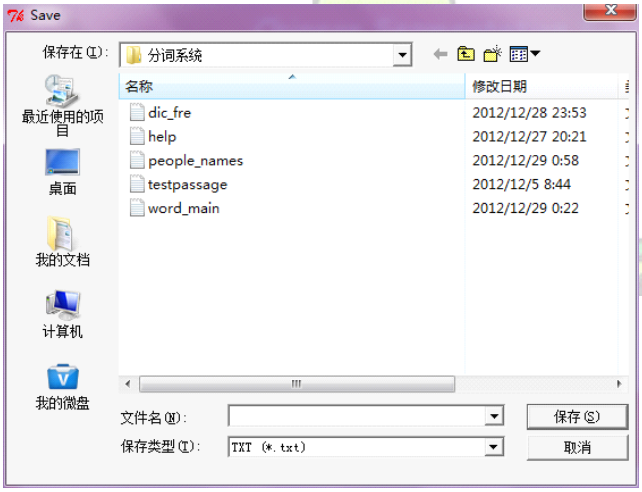
① File:



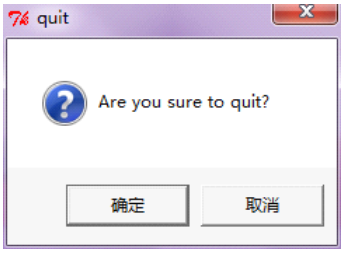
a) Open—open a file whose type is txt and it will be automatically presented. When you click it, the open window will jump out. (In our procedure, no other types of files will be presented in the open window)



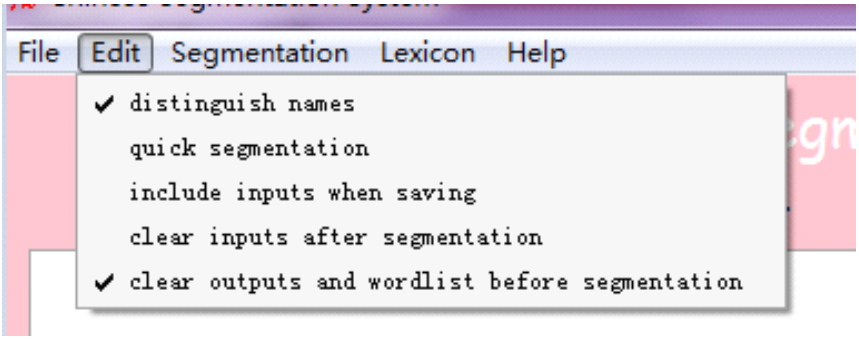
b) Save—save the results in a new text file (*.txt). You can decide whether to include the original text in the Edit menu. When you click this button, the saving window will jump out. (In our procedure, no other types of files will be presented in the save window)



c) Exit—quit the program. When you click this button, a warning box will jump out and you should click “确定” to quit and “取消” to return to the main window.



② Edit: It's made up of a set of check buttons.



a) distinguish names—names will be found out alone if you check this button.

b) quick segmentation—segmentation will only be performed in one order and thus it saves approximately half the time of the formal one if you check this button.

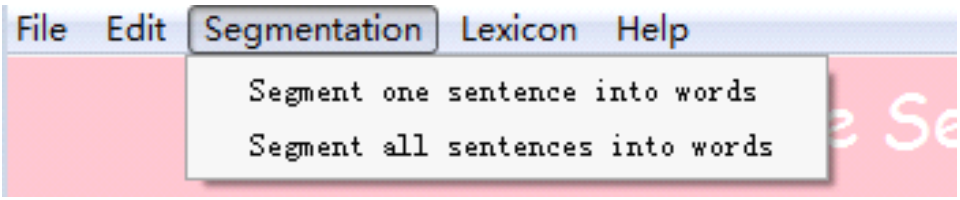
c) Include inputs when saving: the input text will also be saved in the result file if you check this button and only the segmented results will be saved if not.

d) clear inputs after segmentation

e) clear outputs and wordlist before segmentation.

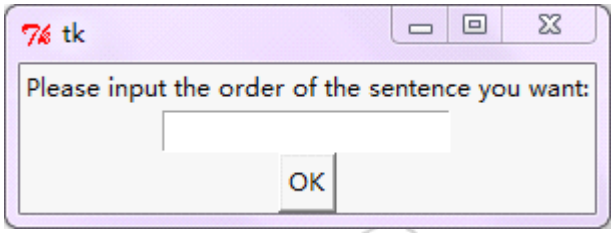
Every time you run it, check buttons—"distinguish names" and "clear outputs and wordlist before segmentation" will be automatically ticked.

③ Segmentation:



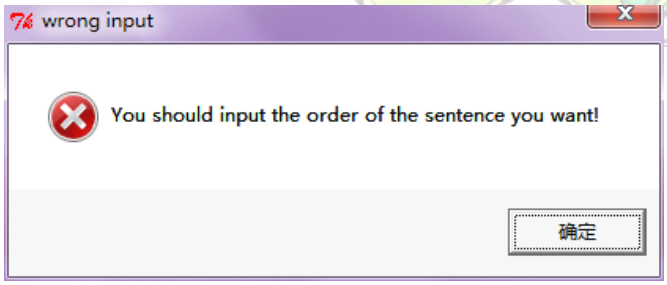
a) Segment one sentence into words

When you click it , a message box will jump out.



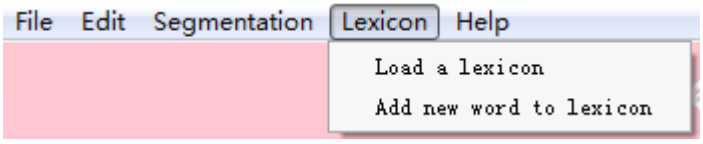
Input the number of the sentences you want to segment alone and click ok to continue.

If the number if out of the range or your input is not a positive integer, a warning box will jump out.



b) Segment all sentences into words

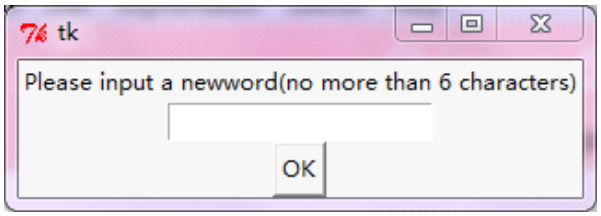
④ Lexicon:



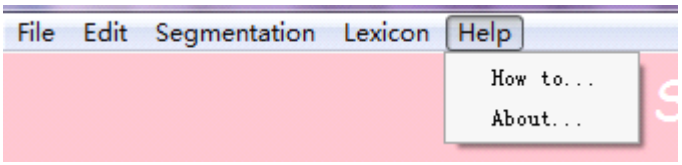
a) Load a lexicon:

b) Add new word to lexicon:

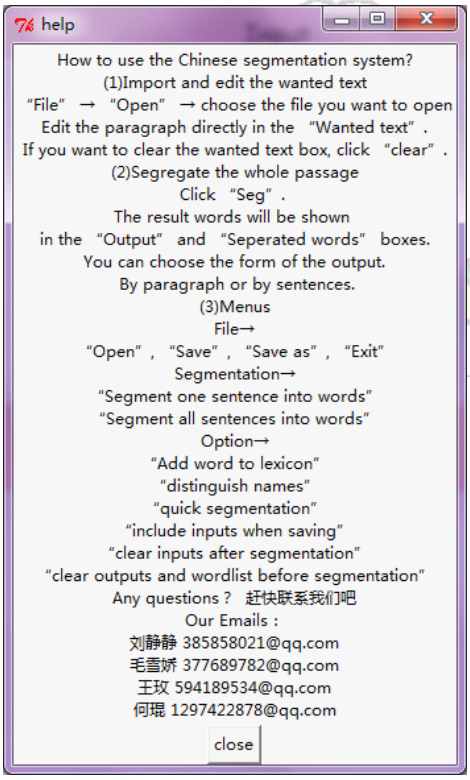
Input the new word in the entry and click OK to save it.



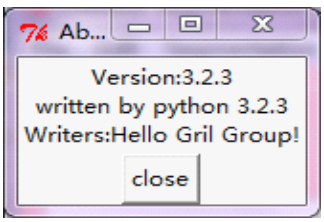
⑤ Help:



a) How to …: click it and you will see the rules.



b) About…: click it and you will see the version.

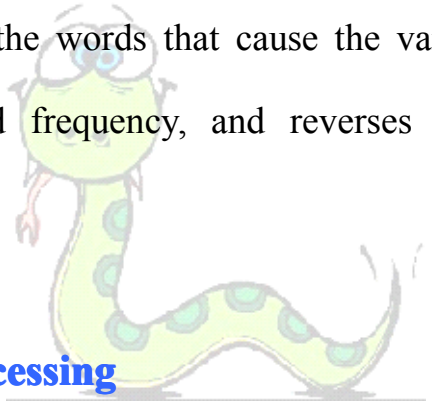


Ⅳ Lexicon Component

§ 4.1 Introduction

To construct the dictionaries, which are the database of the program. When the program reads in the text, it will match the words with the words in the dictionaries. If the words can be found in the dictionary, it will be treated as a single word. Otherwise, it will cut a word and continue to search until the length is one.

We will check both forward and reverse. If the results are different, the project will find the words that cause the various meanings in the dictionary with word frequency, and reverses the one with higher frequency.



§ 4.2 Lexicon processing

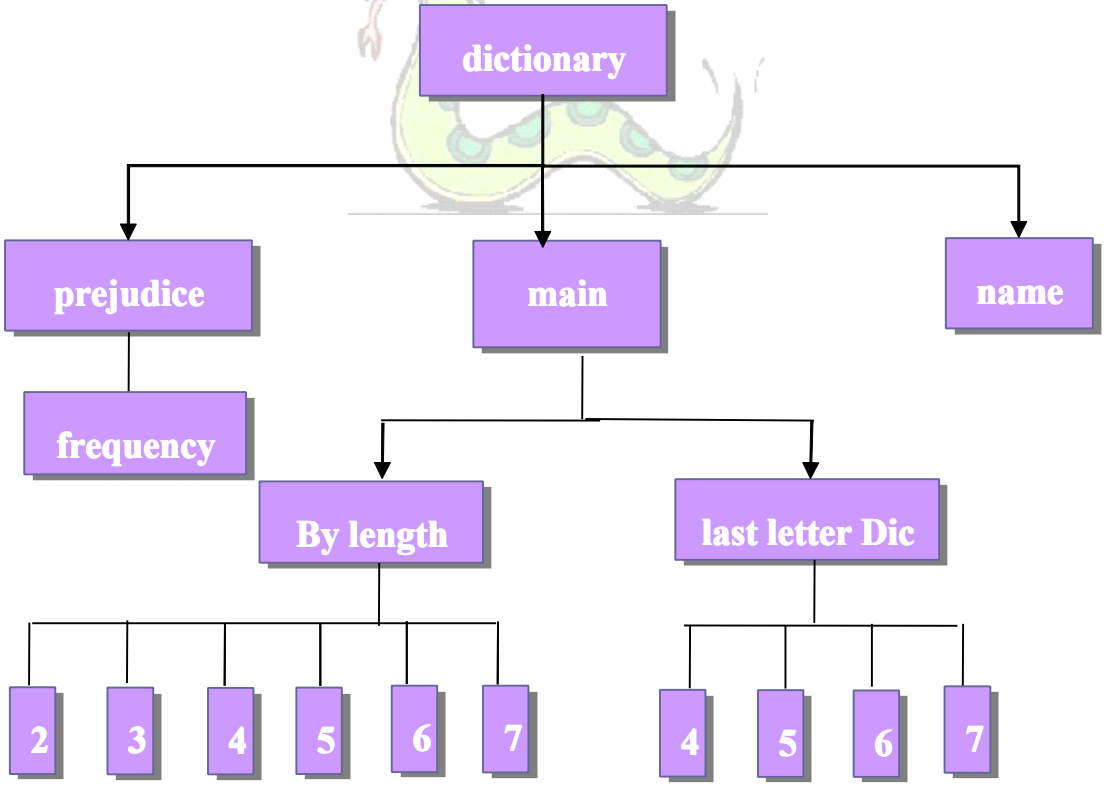
First, we should form a main dictionary, which contains the most words and phrases. After searching the Internet and downloaded many dictionaries, however, many of them are messy codes or have the different version with my Python program, and some has relatively little words. So we combined several dictionaries and preserve the final dictionary as the form of “dic”.

However, later we found the form “dic” can not be read by the program. So we just transformed it into the form “txt” and sort the words

by the length of the words or phrases. When reading through these dictionaries, some words can not be read and the program will occur "Indentation Error: unexpected unindent" because there exists messy codes among the dictionary. So we used the "Try" and "Expect" to ignore these errors. Also, it will jump when it comes to the blank spaces until the eventual line.

The dictionary contains words frequency is arranged according to the frequency from big to small. The dictionary contains the common names also useful when we need to take the people's names points out.

§ 4.3 Lexicon combination



Ⅴ Word Segmentation Component

§ 5.1 Introduction

Suppose you are given a paragraph with Chinese sentences and many other sighs. How can you separate words of each sentences? Which of the algorithms has highest accuracy? Is there many particular cases that you should consider?

§ 5.2 Sentence Segmentation

Our aim is to segment a paragraph containing of many sentences. Punctuations, numbers, letters and other signs are good partitions. So we should segment the paragraph into many pieces first.

{A paragraph divide (para)sentences fmm_separate(sen) or rmm_separate(sen) subsentences}

§ 5.3 Chinese Character String Segmentation

Now we just focus on the subsentences we obtain. Our next step is to segment each of these subsentences.

§ 5.4 Word Segmentation

（string based matching method）

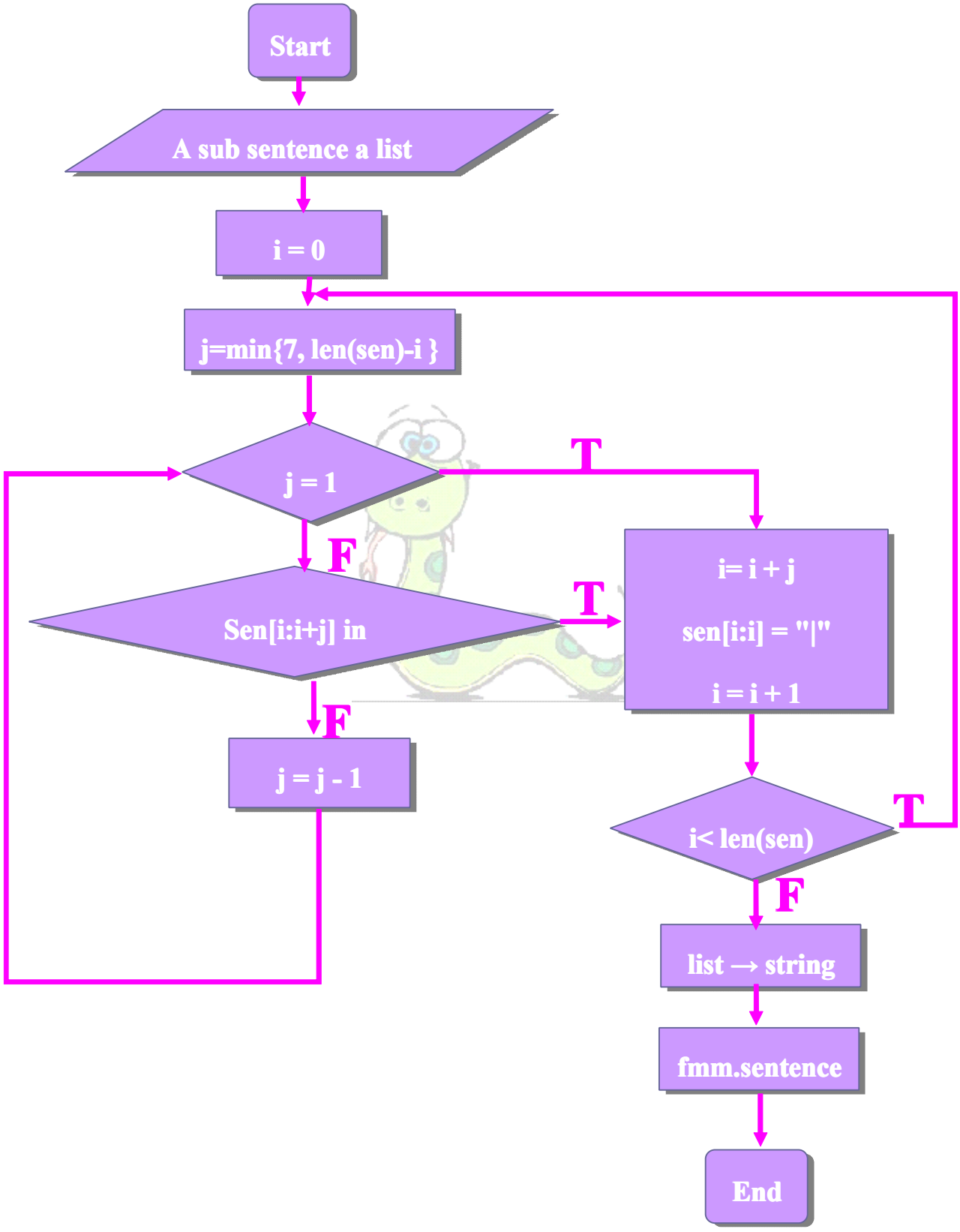
We mainly adopt string based matching method in our program. So a good dictionary is necessary. There are 2 ways in this string based matching method. The match functions use recursive method. At the beginning, the length of the match string is 7. If the matching is success, the function(fmm_match() or rmm_match()) will return the length of the word. Else, length minuses 1 and return the function itself. After receiving the value, another function(fmm_part() or rmm_part()) will add "|" to the end of this word. Because we can match the strings from the sentences with words in dictionary in the forward direction and reverse direction. Then we obtain two results.

（statistics based word segmentation method）

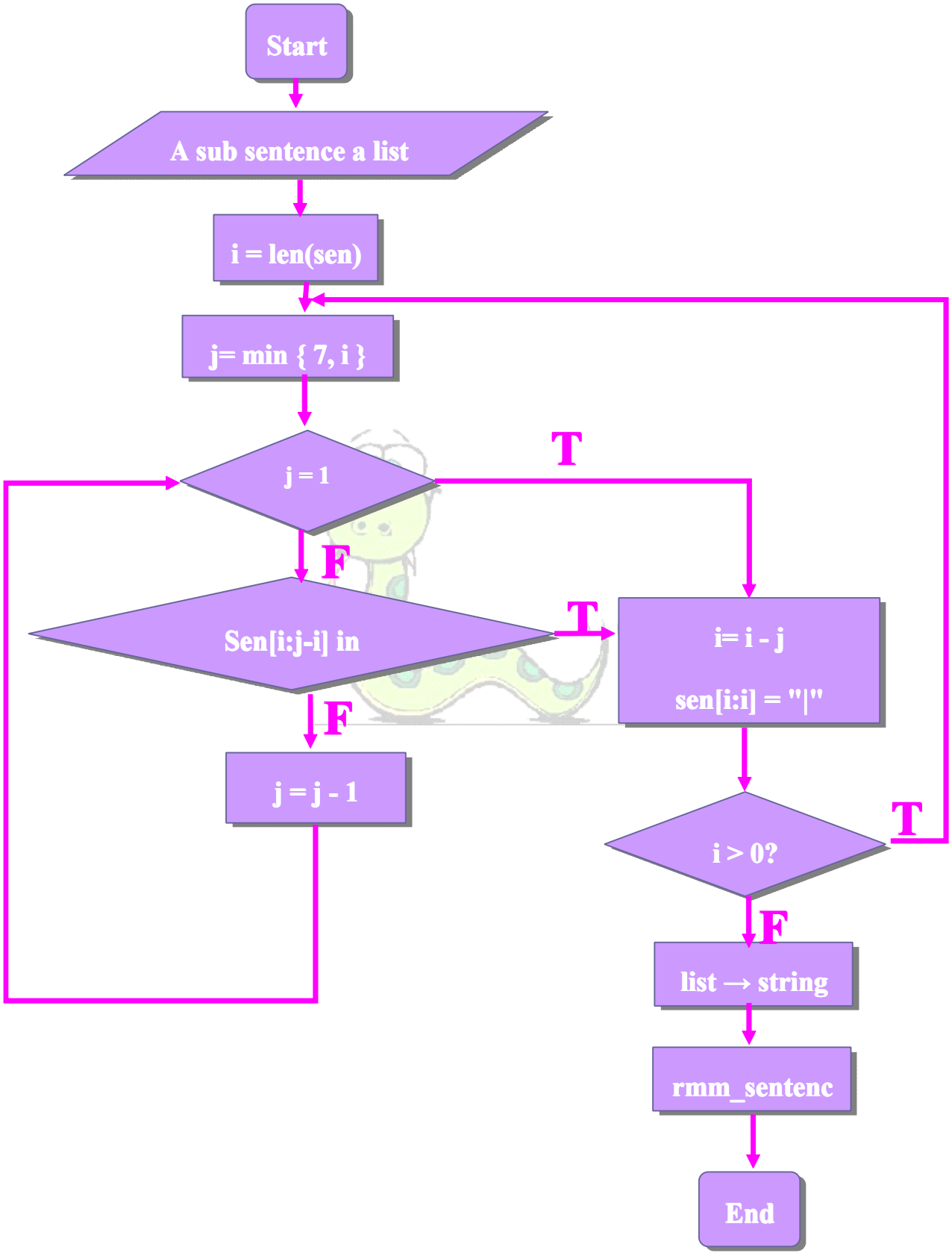
Finally, compare these two results and return the better result. In this step we use statistics based word segmentation method. If two results are same, return. If not, use a function to get the sum of the words' frequences and return the sentence with higher sum. It is more accurate than just segmenting the subsentences from one side. Users can choose whether they need accuracy like this. If you go for speed, you can choose quick segmentation.

§ 5.5 Flowchart

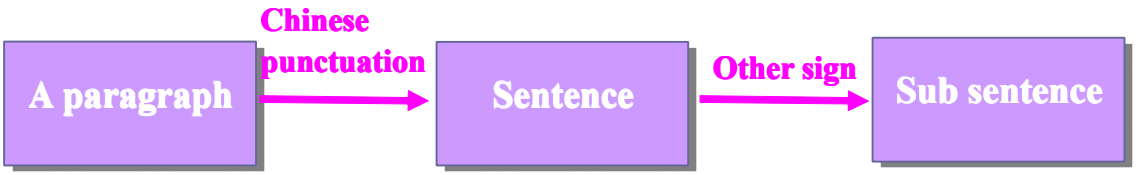
Forward maximum matching method



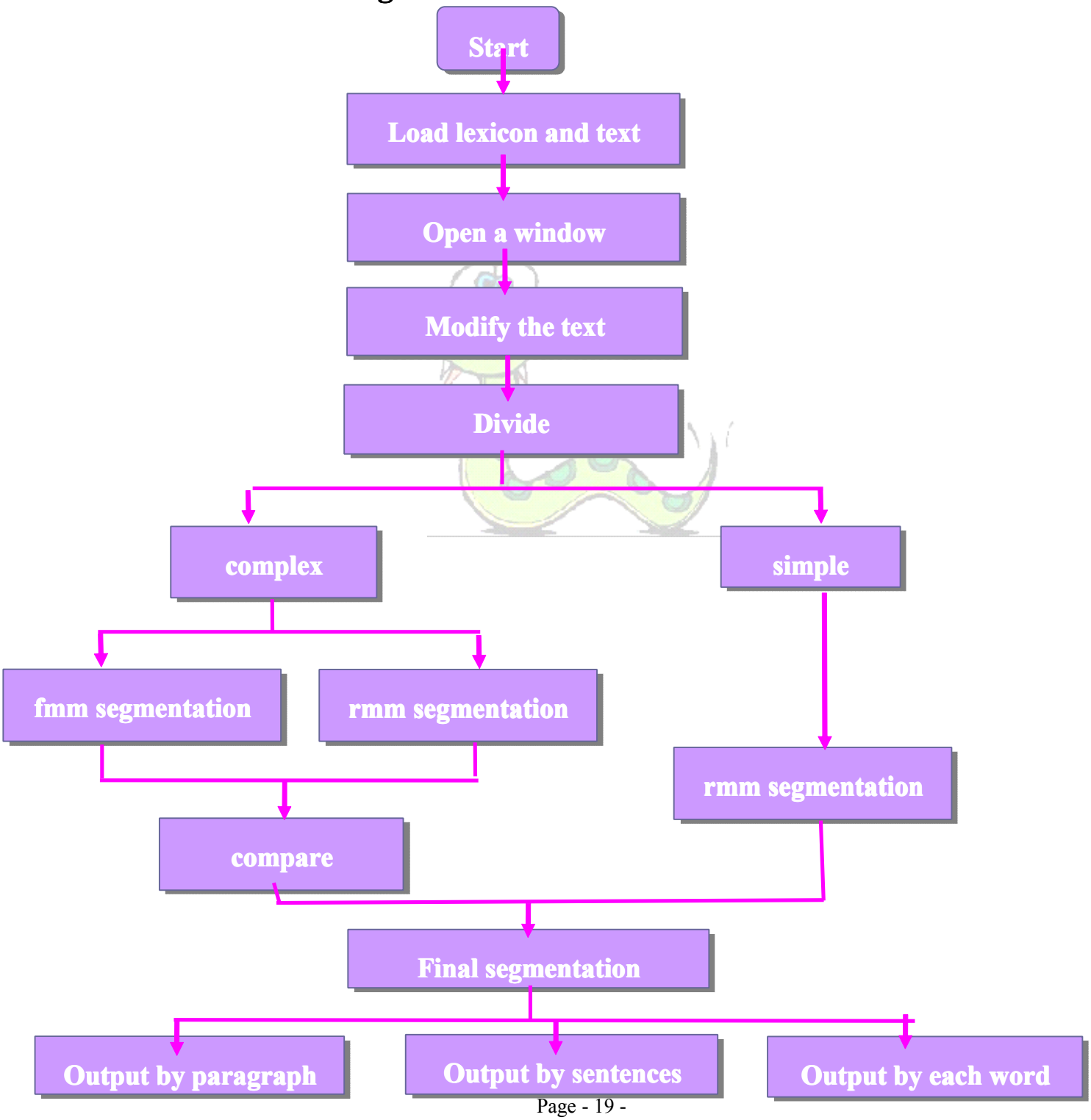
Reverse maximum matching method:



To punctuate:

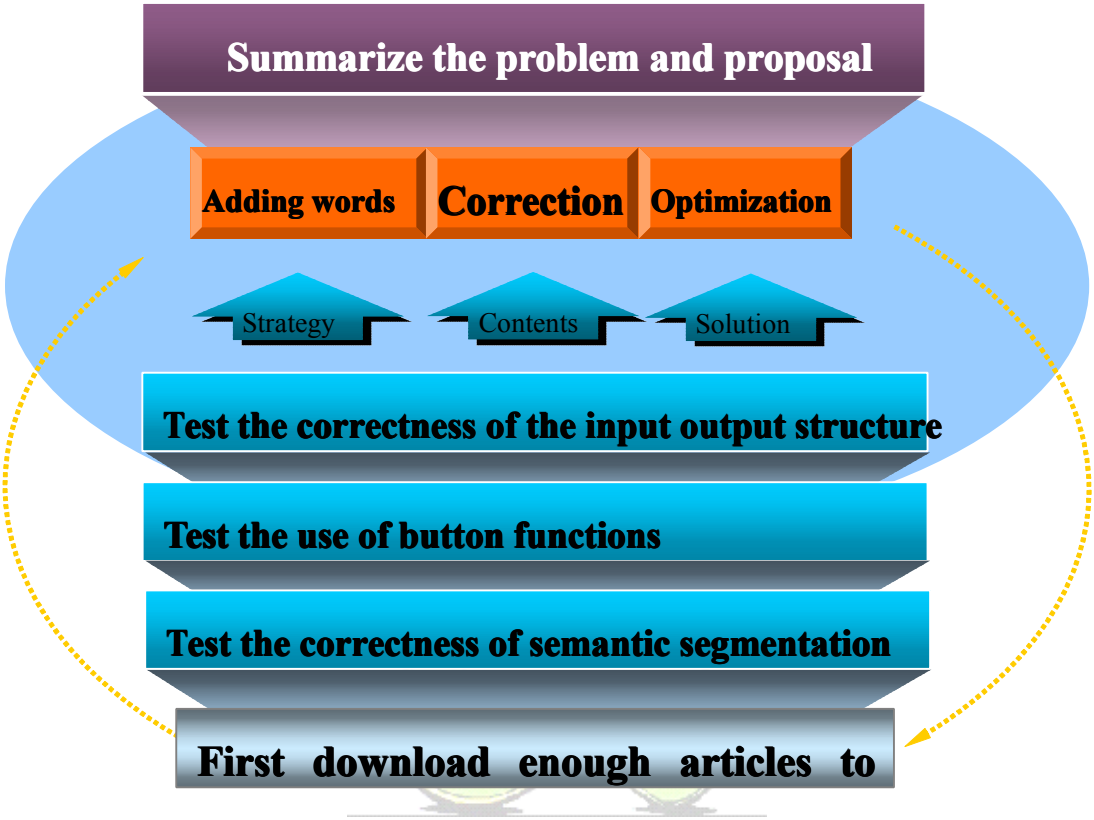


The Whole Program:



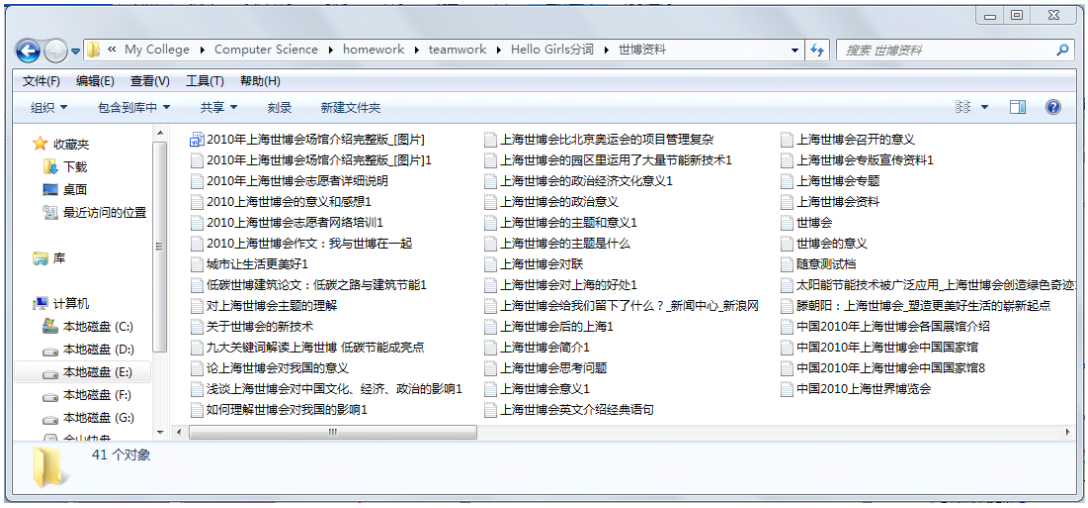
Ⅵ Testing Result

§ 6.1 Main Testing Procedure



§ 6.2 Some Optimization Details

※The download articles to test:



〖The report for the problem and proposal in testing〗

※Test problems on 2012-12-22 ：

1. ”|的|日至|2张|15| 即|2010|年|12|月|1|日至|2011|年|5|月|31|日|。 (“日至”应该分开)
2. |它|鼓励|人类|发挥|创|造|性|和|主动|参与|性| (不该分)
3. |它|更|鼓励|人类|把|科学性|和|情感|结合起来|， (该分)
4. |第一届|真正|意义上|的|世博会|是|1851|年在|英国伦敦|举办|的| (该分)
5. ，‖对|工|业|和|建筑业|影响|和|推动|为|次|， (该分) 而且很多句句话开始时有两个竖杠。。
6. |它|所|包含|的|五|个|分|主题| (不该分)
7. |推动|重大|科技|创|新‖世界博览|会|是|工|业|革命|下|科技进步|的|产物|， (不该分)
8. |推动|了|现|代|科技|进入|人类|生活| (不该分)
9. |实现|经济|资源|的|优化|配|置| (不该分)
10. “|世博|经济|”|现|象|由|三个|部|分|组成|：‖一是|直接| (不该分)
11. |不|仅有|劳动力|、|资本|、|土地|等|有形|要素|，‖而且|涉及到|管理|、 (分错)
12. |具有|多|要素|流动|与|配|置|的|特点|。 (不该分)
13. |在|会|后|阶段|，|主要|表现|为|场馆|设施|的|后续|利|用|对|经济|的|影响|。 (不该分)
14. ‖四|是非|均衡性|。 (分错)
15. |这就|给|主办国|创|造|了|足不出户|便可|宣传|自己|、|广交朋友|的|机|会|。 (分错)

Etc

※Test proposals:

1. I hope it can be more gorgeous. For example: make background a beautiful picture; Start in dynamic way; Adding a background music will be better……

2. The input two module is also needed to be coupled with scroll bar, as when I use the long file to test, it is not convenient....

3. The interface menu format is not as the requirement：

   The final menu format need being as follows：

File: (1) open a file through choosing file name in hard disk or floppy disk; (2) exit from the system.

Edit: (1) edit a file; (2) input a sentence by hand.

Segmentation: (1) segment sentences of a text (input*.txt)by batch processing and output segmented result to another text (output*.txt); (2) segment a sentence into words.

Lexicon: (1) load the lexicon; (2) add a new word into the lexicon.

Rule: (1) load the rule library; (2) add a new rule into the rule library.

Help: (1) instructions about the system; (2) copyright information.

4. The Output format is wrong. We need two choices (0 and 1) about how to divide, if 0, the program now is ok ; if 1, we can choose a sentence to be segregated. Just as the following:

Output1 -Sentence Segmentation:

Would you like to observe the result of sentence segmentation of theabove text (yes = 1 / no = 0)?

Your input: 1

1. 上海世博会中国国家馆12月1日起续展半年
2. 2010年11月25日
3. 世博网11月25日消息：
4. 为满足广大参观者的要求，
5. 让更多公众一睹“东方之冠”的风采，
6. 上海世博会中国国家馆将从12月1日起续展半年，
7. 即2010年12月1日至2011年5月31日。
8. 续展期间，
9. 中国国家馆普通票票价为每张20元，
10. 优惠票每张15元。

Output2 -Word Segmentation:

Would you like to select one of the above sentences to look at the result of word segmentation? If yes, please input the indicated sentence no. If no, please input 0.

Your first input? 1

The result of word segmentation for the 1st sentence:

上海|世博会|中国|国家馆|12|月|1|日|起|续展|半|年|

Your next input? 5

The result of word segmentation for the 5th sentence:

让|更|多|公众|一|睹|“|东方之冠|”|的|风采|，|

Your next input? 0

5. When we close the interface, we`d better say "Thank you for your testing, Goodbye!"

§ 6.3 Data and Result

Testing data:

世博网11月25日消息：为满足广大参观者的要求，让更多公众一睹“东方之冠”的风采，上海世博会中国国家馆将从12月1日起续展半年，即2010年12月1日至2011年5月31日。续展期间，中国国家馆普通票票价为每张20元，优惠票每张15元。

Result table：

|世博网|11|月|25|日|消息|：|为|满足|广大|参观者|的|要求|，|让|更多|公众|一睹|“|东方之冠|”|的|风采|，|上海|世博会|中国国家馆|将从|12|月|1|日|起|续展|半年|，|即|2010|年|12|月|1|日|至|2011|年|5|月|31|日|。|续展|期间|，|中国国家馆|普通票|票价|为|每张|20|元|，|优惠票|每张|15|元|。

§ 6.4 Result Analysis

As for the application in computational linguistics, maximum matching method is mostly widely used. However, our team combine FMM and RMM in our program, which reduces the error rate. What`s more, if there exist differences between them two, we use our frequency dictionary for every word, so that our results have higher accuracy than many other matching methods. To improve its efficiency, some exceptions and names are also added into it, which makes the dictionary more complete and help the segmentation more and more successful.

Ⅶ Conclusion

The application in computational linguistics is always a difficult problem for computer scientists, which makes us need to learn more to solve it correctly. After reading many references, we know that maximum matching method is mostly widely used. So how to improve it becomes our task.

The final goal for any program is to improve its practicability, which includes accuracy rate, time, space, appearance and so on.

To improve the accuracy rate, our team combine FMM and RMM in our program, adding frequency dictionary for every word to reduce the error rate. To improve its efficiency, some exceptions and names are also added into it, which makes the dictionary more complete and help the segmentation more and more successful. To reduce its executing time, we change the matching string into find first word in a dictionary, which achieved remarkable results.

To be honest, we still have many defects in the program, so we have a long way to go in the future. Just as the Professor Yao said, we should not only improve the difficulty ability, but also on the complexity level. There are also many high level method to solve this problem better, such as syntactic analysis, semantic analysis, or even sentence structure analysis and so on. Because of the limits of time and knowledge, we can not do

more at the present.

During solving this team project, we not only learn more knowledge in python programming, but also build up a strong cooperation bond. We allocate the big problem into parts. just as "Divide and Conquer" method in programming, each of us concentrate on our own task, and then combine as well as discuss, learning more and gaining much joy.

We finish the task successfully, which seems a little bit difficult for four girls at the beginning. We are happy at the ending, but we enjoy more during the process.

Thanks!

Reference

[1]. 黄昌宁,赵海, 中文分词十年回顾[J]. 中文信息学报, 2007, (03),pp 8-19.

[2]. 张庆扬,柴胜, 使用二级索引的中文分词词典[J]. 计算机工程与应用, 2009, (19),pp 139-141.

[3]. 赵铁军,吕雅娟,于浩, etc., 提高汉语自动分词精度的多步处理策略[J]. 中文信息学报, 2001, (01),pp 13-18.

[4]. 郑家恒,张剑锋,谭红叶, 中文分词中歧义切分处理策略[J]. 山西大学学报(自然科学版), 2007, (02),pp 163-167.

[5]. John E.Grayson 著, 陈文志, 高垒, 缪瑾, 崔广仁, 蒋涛译, Python与Tkinter编程, 2002, (09).

[6].一种改进的中文分词歧义消除算法研究 许高建 1’2， 胡学钢 2， 路遥 1， 王庆人 1

[7].一种中文分词词典新机制———双字哈希机制! 李庆虎，陈玉健，孙家广